

WHAT IS CLAIMED IS:

1. A method, comprising the steps of:  
providing a token which can be atomically read and which uniquely identifies a log entry which cannot be atomically read and evaluated for change; and  
clearing said log entry using said token as a key.
2. The method according to claim 1 further comprising the steps of:  
storing error data as said log entry and updating said token to correspond to said error data;
3. The method according to claim 1 further comprising the steps of:  
reading said error data using said token to validate said error data.
4. The method according to claim 1 wherein said token includes an indication of an ordinality of said log entry.
5. The method according to claim 1 wherein said token includes an indication of a status of said log entry.
6. The method according to claim 1 further comprising ensuring only valid copies of error data are obtained corresponding to said log entry and inhibiting clearing of unrecorded data corresponding to said log entry.

7. The method according to claim 1 further comprising a step of forming a digital signature of said log entry to create said token.

8. The method according to claim 1 further comprising a step of hashing said log entry to create said token.

9. A method of accessing data, comprising the steps of:

receiving first data;

incrementing a first register containing a count value in response to said first data to provide an incremented count value; and

5 storing, in response to a first condition of a flag, (i) said incremented count value in a second register and (ii) said first data in a memory.

10. The method according to claim 9, further comprising a step of setting said flag to a second condition in response to said first data.

11. The method according to claim 9, further comprising the steps of:

reading a first data value stored in said second register;

reading said first data from said memory;

5 reading a second data value stored in said second register and comparing said first and second data values.

12. The method according to claim 11 further comprising a step of processing said first data in response to a result of said comparing step.

13. The method according to claim 11, further comprising the step of resetting a condition of said flag only if said second data value matches the value stored in said second register.

14. The method according to claim 9, further comprising a plurality of steps of reading portions of said first data from said memory and steps of comparing values read from said second register and, in response, selectively processing said first data.

15. The method according to claim 9, further comprising the steps of:  
setting said flag to a second condition in response to said first data;  
reading, on plural occasions, different portions of said first data from said memory;  
5 comparing values stored in said second register prior to and after said reading step and, in response, selectively processing said first data stored in said memory; and  
in response to said comparing step resetting said flag back to said first condition only if the value read from said second register for use in comparing step matches the value stored in said second register.

16. The method according to claim 15, further comprising the steps of:  
detecting a non-equivalence of said values and, in response, inhibiting a processing of said first data stored in said memory.

17. The method according to claim 9 wherein said first data includes error information and said count value includes a number of error events detected.

18. The method of reading a shared resource in a multiprocessor environment, comprising the steps of:

detecting an event;

incrementing an event count to provide an incremented event count;

5       overwriting old data stored in a memory with new data related to said event in response to an indication that said old data has been processed;

      associating, with said new data related to said event, a reference count corresponding to said incremented event count in response to said indication that said old data has been processed;

10       storing said reference count;

      reading said new data from said memory;

      reading a new copy of said reference count subsequent to said step of reading said new data;

15       comparing said new copy with said stored copy of said reference count and, in response, selectively processing said new data from said memory.

19. The method according to claim 18 further comprising a step of providing an indication that said new data has been processed.

20. The method according to claim 19 further comprising a step of indicating that said new data has been processed by supplying a previously read value of the reference count matching the current value of the reference count.